CLAIMS

1. A method of emulating an Ethernet protocol in a network that uses a non-Ethernet protocol, comprising the steps of:

(1) converting a first request to transmit a packet using an Ethernet protocol into a second request to transmit the packet using the non-Ethernet protocol, wherein the conversion is based on a stored mapping between Ethernet and non-Ethernet addresses;

(2) encapsulating the packet into a second packet compatible with the non-Ethernet protocol; and

(3) transmitting the second packet over the network to a destination node in the network.

2. The method of claim 1, further comprising the step of, prior to step (1), generating an Ethernet address derived from node addressing information.

3. The method of claim 1, further comprising the step of, in response to determining that no mapping exists for a particular Ethernet address, transmitting the second packet over a bridge channel that is monitored by a network bridge.

4. The method of claim 2, wherein the generating step comprises the step of using a digital signature algorithm to generate the Ethernet address.

5. The method of claim 4, wherein the generating step comprises the step of performing an exclusive OR operation between a portion of a digital signature generated by the digital signature algorithm and a stored constant.

6. The method of claim 2, further comprising the step of determining whether the generated Ethernet address conflicts with other Ethernet addresses in the network.

7. The method of claim 6, wherein the determining step comprises the step of transmitting the generated Ethernet address and the node addressing information to other nodes in the network and evaluating responses from the other nodes.

8. The method of claim 1, wherein step (1) comprises the step of binding an Ethernet-compliant application programming interface (API) that emulates Ethernet functionality using an IEEE 1394 serial bus.

9. The method of claim 1, wherein step (1) comprises the step of converting the first request to transmit the packet into an IEEE-1394 serial bus request, and wherein step (3) comprises the step of transmitting the second packet over an asynchronous data channel on the serial bus.

10. The method of claim 1, wherein step (1) comprises the step of converting the first request to transmit the packet into an IEEE-1394 serial bus request, and wherein step (3) comprises the step of transmitting the second packet over an isochronous data channel on the serial bus.

11. The method of claim 1, further comprising the step of determining an optimum packet size for use in transmitting packets over the network.

12. The method of claim 11, wherein the step of determining the optimum packet size is performed by consulting speed topology maps for heterogeneous nodes on the network.

13. The method of claim 12, wherein the optimum packet size is selected based on a maximum packet size that can be accommodated by any two Ethernet-emulated nodes on the network.

14. A system that supports Ethernet emulation, comprising:

a non-Ethernet network communication medium; and

a plurality of network nodes coupled to the non-Ethernet network communication medium, each network node comprising in combination

a network card and associated drivers that transmit and receive packets over the network communication medium using network-specific protocols;

a Transmission Control Protocol/Internet Protocol (TCP/IP) protocol;

an Address Resolution Protocol (ARP) that translates IP addresses into Ethernet Media Access Control (MAC) addresses; and

an interface driver that presents an Ethernet-compliant interface to the TCP/IP protocol and ARP;

wherein the interface driver converts Ethernet-specific requests from the TCP/IP protocol and ARP into requests that conform to the network-specific protocols.

15. The system according to claim 14, wherein the network communication medium comprises an IEEE-1394 compliant serial bus; and wherein the interface driver converts requests to transmit Ethernet packets into requests to transmit 1394 bus-specific packets.

16. The system according to claim 15, wherein the interface driver, in response to determining that a destination Ethernet address is unknown, transmits a packet over a bridge channel that is monitored by a network bridge.

17. The system according to claim 14, wherein the interface driver receives node-specific information from other network nodes and, in response thereto, associates the node-specific information with Ethernet MAC addresses.

18. The system according to claim 15, wherein the interface driver generates an internal table that maps an Ethernet address to node addressing information obtained from another node in the network and stores the mapping information for use in transmitting data packets over the communication medium.

19. The system according to claim 18, wherein the interface driver generates an Ethernet address using a digital signature algorithm applied to the node addressing information.

20. The system according to claim 18, wherein the interface driver performs a conflict resolution procedure to determine whether the generated Ethernet address conflicts with other Ethernet addresses in the system.

21. The system according to claim 15, wherein the interface driver transmits emulated Ethernet packets using an asynchronous channel on the 1394 bus.

22. The system according to claim 15, wherein the interface driver transmits emulated Ethernet packets using an isochronous channel on the 1394 bus.

23. The system according to claim 14, further comprising a network bridge node coupled to the network communication medium, wherein in response to a unicast Ethernet packet received on the bridge channel, the network bridge node retransmits the packet to other networks to which the network bridge node is connected.

24. The system according to claim 14, wherein the interface driver determines an optimum packet size for transmitting emulated Ethernet packets on the network communication medium.

25. The system according to claim 24, wherein the interface driver determines the optimum packet size by consulting speed topology maps from a plurality of network nodes.